Claims

1. A file system, having a plurality of storage blocks, and including a plurality of bits associated with each one of said plurality of storage blocks, at least one of said plurality of bits identifying whether said one storage block was part of said file system at a time earlier than a current consistent version of said file system.

2. A file system as in claim 1, including a second one of said plurality of bits identifying whether said one storage block was part of said file system at a second time earlier than a current consistent version of said file system

3. A file system as in claim 2, including an element disposed for selecting storage blocks in response to said one bit and said second one bit associated with said selected storage blocks.

4. A file system as in claim 3, including an element disposed for copying said selected storage blocks to a destination.

5. A file system as in claim 4, wherein said destination includes: a tape, a disk, a data structure in a second file system, a set of network messages, or a destination distributed over a plurality of file systems.

6. A file system as in claim 1, including an element disposed for selecting storage blocks in response to said one bit associated with said selected storage blocks.

7. A file system as in claim 6, including an element disposed for copying said selected storage blocks to a destination.

8. A file system as in claim 7, wherein said destination includes: a tape, a disk, a data structure in a second file system, a set of network messages, or a destination distributed over a plurality of file systems.

9. A file system having a plurality of storage blocks, said file system including a snapshot including a set of member storage blocks selected from said plurality, said member storage blocks forming a consistent file system other than an active file system; said snapshot being disposed as an object in said file system, wherein said file system is responsive to at least one file system request with regard to said snapshot.

10. A file system as in claim 9, including

a shadow snapshot of a set of member storage blocks selected from said plurality, said member storage blocks having formed a consistent file system other than an active file system, with a set of selected member storage blocks removed from said consistent file system; and

a storage image defined in response to said snapshot and said shadow snapshot, said storage image indicating a set of member storage blocks selected from said plurality.

11. A file system as in claim 9, including a plurality of said snapshots; wherein said plurality of said snapshots are associated with an array of bits, said array having one set of bits for each storage block in said plurality of storage blocks, said set of bits having at least one bit for each said snapshot.

12. A file system as in claim 9, wherein said file system can manipulate said snapshot without having to traverse a hierarchy of file system objects within said snapshot.

13. A file system as in claim 9, wherein said snapshot includes a data structure disposed in a format allowing for a set management operation to be performed efficiently.

14. A file system as in claim 9, wherein said snapshot includes an array of bits, said array having one bit for each storage block in said plurality.

15. A file system as in claim 9, including

a plurality of said snapshots; and

a storage image determined in response to said plurality of snapshots;

said storage image defining a second set of member storage blocks selected from said plurality.

16. A file system as in claim 15, wherein said storage image is a result of a set management operation on said set of member storage blocks for said snapshot.

17. A file system as in claim 9, wherein said snapshot includes a data structure disposed in a format allowing for a set management operation to be performed in O(n) time or less, where n is a number of storage blocks in said plurality, without reading any contents of said storage blocks in said plurality.

18. A file system as in claim 17, wherein said set management operation is a logical sum or difference.

19. A file system as in claim 9, wherein said snapshot includes a data structure identifying which storage blocks in said plurality are member storage blocks of said snapshot.

20. A file system as in claim 19, wherein said data structure uses no more than $1/100^{th}$ of an amount of storage required by said storage blocks in said plurality.

21. A file system as in claim 19, wherein said data structure uses no more than four bytes per storage block in said plurality.

22. A method to be performed in a file system, said file system having a plurality of storage blocks, said method including steps for

defining a storage image of a set of member storage blocks selected from said plurality, said storage image being formed based on a set of member storage blocks forming a consistent file system other than an active file system; and

forming an image stream of a sequence of member storage blocks selected from said storage image.

23. A method as in claim 22, including steps for associating a block location with each one of said sequence.

24. A method as in claim 22, further including steps for reconstructing a file system based on said image stream.

25. A method as in claim 22, including repeating said defining step at periodic intervals.

26. Apparatus including

a file system including a plurality of snapshots thereof, each representing an associated consistent state at an associated selected time; and

each said snapshot including an indication of a set of storage blocks in said associated consistent state, said indication being recorded in at least one storage block in said associated consistent state.

27. In a file system having a plurality of storage blocks, a data structure including

a first snapshot of a set of member storage blocks selected from said plurality, said member storage blocks forming a consistent file system other than an active file system;

said first snapshot being represented as an object in said file system and having a set of storage blocks for recording said first snapshot;

whereby copying said member storage blocks in said first snapshot has the property of preserving at least one snapshot recorded in said file system at a time of said first snapshot.

28. A data structure as in claim 27, including

a second snapshot of a set of member storage blocks selected from said plurality, said member storage blocks forming a consistent file system other than an active file system;

said second snapshot being represented as an object in said file system and having a set of storage blocks for recording said second snapshot;

whereby copying said member storage blocks in said second snapshot has the property of preserving at least one snapshot recorded in said file system at a time of said second snapshot.

29. A data structure as in claim 27, including

an image stream including a set of storage blocks including both said first snapshot and said second snapshot;

whereby copying said member storage blocks in said image stream has the property of preserving both said first snapshot and said second snapshot.

30. In a file system having a plurality of storage blocks, a data structure including

a snapshot of a set of member storage blocks selected from said plurality, said member storage blocks forming a consistent file system other than an active file system;

said snapshot being represented as an object in said file system and having a set of storage blocks for recording said snapshot;

whereby a backup and restore operation on said file system has the property of preserving said snapshot within said file system.

31. In a file system having a plurality of storage blocks, a data structure including

a storage image of a set of member storage blocks selected from said plurality;

said storage image being formed based on a set of member storage blocks forming a consistent file system other than an active file system.

32. A data structure as in claim 31, including

a first storage image indicating a set of member storage blocks forming a consistent file system; and

a sequence of incremental storage images, each having a predecessor, at least one of said predecessors being said first storage image;

wherein a logical sum of said set of storage images includes at least one complete snapshot.

33. A data structure as in claim 31, wherein said storage image indicates a set of member storage blocks forming a consistent file system.

34. In a file system having a plurality of storage blocks, a data structure stored in said file system, including a shadow snapshot of a set of member storage blocks selected from said plurality, said member storage blocks having formed a consistent file

system other than an active file system, with a set of selected member storage blocks removed from said consistent file system.

35. A data structure as in claim 34, wherein said shadow snapshot uses, in addition to said member storage blocks, no more than 1/100th of an amount of storage required by said storage blocks in said plurality.

36. A data structure as in claim 34, wherein said shadow snapshot is disposed as a single object in said file system, whereby said file system can manipulate said snapshot without having to traverse a hierarchy of file system objects within said snapshot.

37. A data structure as in claim 34, wherein said removed member storage blocks are responsive to completion of a processing operation.

38. A data structure as in claim 37, wherein said processing operation includes a file system operation.

39. A data structure as in claim 37, wherein said processing operation includes reuse of said selected member storage blocks by said file system.

40. A data structure as in claim 34, wherein said shadow snapshot is disposed in a format allowing for a set management operation to be performed in O(n) time or less, where n is a number of storage blocks in said plurality, without reading any contents of said storage blocks in said plurality.

41. In a file system having a plurality of storage blocks, a data structure stored in said file system, including a mark-on-allocate image of a set of member storage blocks selected from said plurality, said member storage blocks having been added to a snapshot that originally formed a consistent file system.

42. A data structure as in claim 41, wherein said mark-on-allocate storage image is disposed as a single object in said file system, whereby said file system can manipulate said snapshot without having to traverse a hierarchy of file system objects within said snapshot.

43. A data structure as in claim 41, wherein said mark-on-allocate image is disposed in a format allowing for a set management operation to be performed efficiently.

44. A data structure as in claim 41, wherein said mark-on-allocate storage image uses no more than 1/100th of an amount of storage required by said storage blocks in said plurality.

45. A data structure as in claim 41, said member storage blocks having been selected responsive to completion of a processing operation.

46. A data structure as in claim 45, wherein said processing operation includes a file system operation.

47. A data structure as in claim 45, wherein said processing operation includes reuse of said selected member storage blocks by said file system.

48. A data structure as in claim 41, wherein said mark-on-allocate image is disposed in a format allowing for a set management operation to be performed in O(n) time or less, where n is a number of storage blocks in said plurality, without reading any contents of said storage blocks in said plurality.

49. In a file system having a plurality of storage blocks, a data structure stored in said file system, including a mark-on-deallocate image of a set of member storage blocks selected from said plurality, said member storage blocks having been removed from a snapshot that originally formed a consistent file system.

50. A data structure as in claim 49, wherein said mark-on-deallocate storage image uses no more than 1/100th of an amount of storage required by said storage blocks in said plurality.

51. A data structure as in claim 49, wherein said mark-on-deallocate image is disposed in a format allowing for a set management operation to be performed in O(n) time or less, where n is a number of storage blocks in said plurality, without reading any contents of said storage blocks in said plurality.

52. A method for recording a plurality of data about a plurality of blocks of data stored in storage means, comprising the steps of:

maintaining a means for recording multiple usage bits per block of said storage means; and

storing, in said means for recording multiple usage bits per block, multiple bits for each of said plurality of said blocks of said storage means, at least one of said multiple bits being indicative of block reusability.

53. A method for recording a plurality of data about a plurality of blocks of stored data, comprising the steps of:

recording multiple usage bits per block of said stored data; and

storing multiple bits for each of said plurality of said blocks of stored data, at least one of said multiple bits being indicative of block reusability.